WE CLAIM:

1. A computer system comprising a plurality of processor clusters interconnected by a plurality of point-to-point inter-cluster links, each processor cluster comprising nodes including a plurality of local processors and an interconnection controller interconnected by a plurality of point-to-point intra-cluster links, communications within a cluster being made via an intra-cluster protocol that uses intra-cluster packets, wherein the interconnection controller in each cluster is operable to map locally-generated communications directed to others of the clusters to the point-to-point inter-cluster links and to map remotely-generated communications directed to the local nodes to the point-to-point intra-cluster links, communications between clusters being made via an inter-cluster protocol that uses inter-cluster packets, an inter-cluster packet encapsulating at least one intra-cluster packet, each interconnection controller configured to compute a first cyclic redundancy code check for, and to encode first cyclic redundancy code check data in, each inter-cluster packet transmitted on the point-to-point inter-cluster links.

2. The computer system of claim 1, wherein each interconnection controller is further configured to compute a second cyclic redundancy code check for each inter-cluster packet received on the point-to-point inter-cluster links.

3. The computer system of claim 1, wherein each interconnection controller is further configured to transmit each inter-cluster packet as an indivisible unit on the point-to-point inter-cluster links.

4. The computer system of claim 1, wherein each interconnection controller is further configured to encode a sequence identifier in each inter-cluster packet transmitted on the point-to-point inter-cluster links.

5. The computer system of claim 1, wherein the nodes are configured to compute a cyclic redundancy code check on a transmission window that includes transmissions of multiple intra-cluster packets on the point-to-point intra-cluster links.

6. The computer system of claim 1, wherein each interconnection controller is further configured to compute a cyclic redundancy code check on a transmission window that includes transmissions of multiple intra-cluster packets on the point-to-point intra-cluster links.

7. The computer system of claim 1, wherein each interconnection controller encodes the cyclic redundancy code check data in a field reserved for a link layer of each inter-cluster packet transmitted on the point-to-point inter-cluster links.

8. A computer system, comprising:

a first cluster including a first plurality of processors and a first interconnection controller, the first plurality of processors and the first interconnection controller interconnected by first point-to-point intra-cluster links; and

a second cluster including a second plurality of processors and a second interconnection controller, the second plurality of processors and the second interconnection controller interconnected by second point-to-point intra-cluster links, the first interconnection controller coupled to the second interconnection controller by point-to-point inter-cluster links;

wherein the first interconnection controller is configured to:

receive a request according to an intra-cluster protocol from a first processor in the first plurality of processors;

generate a probe packet responsive to the request;

compute a first cyclic redundancy code check based only upon bits in the probe packet;

encode first cyclic redundancy code check data in the probe packet; and

send the probe packet to the second interconnection controller in the second cluster.

9. The computer system of claim 8, wherein the first interconnection controller is further configured to encode a sequence identifier in the probe packet.

10. The computer system of claim 8, wherein the second interconnection controller is configured to compute a second cyclic redundancy code check based only upon bits in the first probe packet.

11. The computer system of claim 8, wherein the second interconnection controller is configured to forward the probe packet according to the intra-cluster protocol to a processor in the second plurality of processors.

12. The computer system of claim 8, wherein the second interconnection controller is configured to forward the probe packet according to the intra-cluster protocol to each processor in the second plurality of processors.

13. The computer system of claim 11, wherein the processor in the second plurality of processors is configured to send a response packet according to the intra-cluster protocol to the second interconnection controller.

14. The computer system of claim 12, wherein each processor in the second plurality of processors is configured to send a response packet according to the intra-cluster protocol to the second interconnection controller.

15. An interconnection controller, comprising:

an intra-cluster interface configured for coupling with intra-cluster links to a plurality of local processors arranged in a point-to-point architecture in a local cluster;

an inter-cluster interface configured for coupling with an inter-cluster link to a non-local interconnection controller in a non-local cluster;

a transceiver configured to receive an intra-cluster packet from a local processor via an intra-cluster link and encode a sequence identifier in a header of the intra-cluster packet; and

a serializer/deserializer configured to serialize the encoded packet and forward the encoded, serialized packet to the inter-cluster interface for transmission to the non-local interconnection controller via an inter-cluster link.

16. The interconnection controller of claim 15, wherein the transceiver is further configured to compute a cyclic redundancy code check based only on the encoded packet and to encode the cyclic redundancy code check in the encoded packet.

17. The interconnection controller of claim 15, wherein the inter-cluster interface is further configured to receive encoded, serialized packets from the non-local interconnection controller, wherein the serializer/deserializer is further configured to deserialize the encoded, serialized packets and wherein the transceiver is further configured to perform a cyclic redundancy code check on the deserialized packets.

18. An integrated circuit comprising the interconnection controller of claim 15.

19. A set of semiconductor processing masks representative of at least a portion of the interconnection controller of claim 15.

20. At least one computer-readable medium having data structures stored therein representative of the interconnection controller of claim 15.

21. The integrated circuit of claim 18, wherein the integrated circuit comprises an application-specific integrated circuit.

22. The at least one computer-readable medium of claim 20, wherein the data structures comprise a simulatable representation of the interconnection controller.

23. The at least one computer-readable medium of claim 20, wherein the data structures comprise a code description of the interconnection controller.

24. The at least one computer-readable medium of claim 22, wherein the simulatable representation comprises a netlist.

25. The at least one computer-readable medium of claim 23, wherein the code description corresponds to a hardware description language.

26. A computer-implemented method for detecting errors in a computer system comprising a plurality of clusters, each cluster including a plurality of local nodes and an interconnection controller interconnected by point-to-point intra-cluster links, communications between the local nodes and the interconnection controller made via an intra-cluster protocol using intra-cluster packets, the interconnection controller of each cluster interconnected by point-to-point inter-cluster links with the interconnection controller of other clusters, the computer-implemented method comprising:

forming an inter-cluster packet by encapsulating an intra-cluster packet;

encoding a sequence identifier in the inter-cluster packet;

calculating first cyclic redundancy code check data based only upon the inter-cluster packet;

encoding the first cyclic redundancy code check data in the inter-cluster packet; and

transmitting the inter-cluster packet from a first interconnection controller to a second interconnection controller on a point-to-point inter-cluster link.

27. The computer-implemented method of claim 26, wherein the encoding steps comprise encoding in an area of the inter-cluster packet reserved for link layer information.

28. The computer-implemented method of claim 26, further comprising:

receiving the inter-cluster packet; and

calculating second cyclic redundancy code check data based only upon the inter-cluster packet.

29. The computer-implemented method of claim 28, further comprising:

detecting an error in the inter-cluster packet based upon the second cyclic redundancy code check data; and

notifying the first interconnection controller of the error.

30. An apparatus for detecting errors in a computer system comprising a plurality of clusters, each cluster including a plurality of local nodes and an interconnection controller interconnected by point-to-point intra-cluster links, communications between the local nodes and the interconnection controller made via an intra-cluster protocol using intra-cluster packets, the interconnection controller of each cluster interconnected by point-to-point inter-cluster links with the interconnection controller of other clusters, the apparatus comprising:

means for forming an inter-cluster packet by encapsulating an intra-cluster packet;

means for encoding a sequence identifier in the inter-cluster packet;

means for calculating first cyclic redundancy code check data based only upon the inter-cluster packet;

means for encoding the first cyclic redundancy code check data in the inter-cluster packet; and

means for transmitting the inter-cluster packet from a first interconnection controller to a second interconnection controller on a point-to-point inter-cluster link.